

Peter beim Graben, Heiner Drenhaus

Computationelle Neurolinguistik

Abstract: Computational neurolinguistics integrates methods from computational (psycho-)linguistics and computational neuroscience in order to model neural correlates of linguistic behavior. We illustrate these techniques using an example of the language processing of German negative polarity items (NPI) in the event-related brain potential (ERP) paradigm. To that aim, we first describe the syntactic and semantic licensing conditions of NPIs by means of slightly modified minimalist grammars. In a second step we use dynamic cognitive modeling (DCM) to map the state descriptions of a minimalist parser onto activation patterns of a neural network. Thirdly, the network's synaptic weights are trained with the correct parse of NPI constructions. Using these weights we calculate neural harmony measures for correct and for ungrammatical NPI constructions. In a final step we correlate the harmonies of the dynamical model with experimentally obtained ERP amplitudes by means of a simple statistical model.

Peter beim Graben: Institut für Deutsche Sprache und Linguistik und Bernstein Center for Computational Neuroscience, Humboldt-Universität zu Berlin, Unter den Linden 6, D-10099 Berlin, E-Mail: peter.beim.graben@hu-berlin.de

Heiner Drenhaus: Psycholinguistik, Institut für Computerlinguistik und Phonetik, Universität des Saarlandes, D-66123 Saarland, E-Mail: drenhaus@coli.uni-saarland.de

- 1 Einleitung
- 2 Methoden
 - 2.1 Negative Polaritäts-Elemente
 - 2.1.1 Minimalistische Syntax
 - 2.1.2 Ereigniskorrelierte Gehirnpotenziale
 - 2.2 Neuronale Netzwerkmodelle
 - 2.2.1 Allgemeine Übersicht
 - 2.2.2 Modell-Implementation
 - 2.2.2.1 Fraktale Tensorprodukt-Darstellung
 - 2.2.2.2 Netzwerk-Training
 - 2.2.2.3 Harmonie-Berechnung
 - 2.2.2.4 Statistische Modellierung
- 3 Ergebnisse
- 4 Diskussion
 - Danksagung
 - Anhang
 - Literatur

1 Einleitung

Die „Computermetapher des Geistes“, nach welcher kognitive Prozesse Berechnungen in einem symbolverarbeitenden Automaten, i. a. in einer universellen Turing-Maschine (Turing 1950), seien, zählt zu den erfolgreichsten Annahmen in den Kognitionswissenschaften. In der kognitiven Psychologie konnten mit ihrer Hilfe u. a. Problemlösungs-Strategien erfolgreich modelliert werden (für eine Übersicht s. z. B. Anderson (1996)). In der Computerlinguistik gehören Automatentheorie und formale Sprachen zum Handwerkszeug für Modellierung und für Anwendungsprogramme (Carstensen et al. 2010, Hopcroft & Ullman 1994), und auch in der computationellen Psycholinguistik (Crocker 1996, Lewis 2003) leisten automaten-theoretische Modelle, wie z. B. deterministische Parser, wichtige Dienste zur Aufklärung psycholinguistischer Sachverhalte, z. B. von Holzweg-Sätzen (Pereira 1985).

Ungeachtet ihres Nutzens verleitet die „Computermetapher des Geistes“ leider zu dem Kategorienfehler, auch das menschliche Gehirn metaphorisch als Digitalcomputer anzusehen. Dieser Fehlschluss kann leicht vermieden werden, wenn man sich Dennetts Unterscheidung zwischen der *funktionalen Einstellung* (*design stance*) und der *physikalischen Einstellung* (*physical stance*) zu Eigen macht (Dennett 1978). Ein Digitalcomputer ist einerseits ein Artefakt, das einer bestimmten Funktion dienend für diese Funktion gestaltet worden ist. Andererseits ist der Computer physikalisch *realisiert* (oder *implementiert*)¹ in Form hochkomplexer Netzwerke elektronischer Schalter, sogenannter Transistoren, die vom Standpunkt der Theorie dynamischer Systeme (Jäger 1996, Kaplan & Glass 1995) betrachtet, nichtlineare Bauelemente sind. Dementsprechend ist das Gehirn, aus der physikalischen Einstellung gesehen, kein Digitalcomputer, sondern ebenfalls ein komplexes, nicht-lineares dynamisches System, das sich aus Nervenzellen zusammensetzt.

Auch der dynamische System-Ansatz hat eine gewisse Tradition in den Kognitionswissenschaften (Kelso 1995, Jäger 1996, Schnelle 1996). Hier wird das Problem einer kognitiven Berechnung als *Anfangsbedingung* in einem *Zustandsraum* präpariert, aus der dynamisch eine *Bahnkurve* oder *Trajektorie* generiert wird. Diese Bahnkurve verbindet *transient* Zwischenergebnisse, die häufig als sogenannte *Sattelpunkte* modelliert werden (Rabinovich et al. 2008), bis sie schließlich in einen *Attraktor* mündet, der das Endergebnis repräsentiert.² Der

¹ Zum vermeintlichen Unterschied zwischen Realisation und Implementation s. Smolensky (1988), beim Graben (2004).

² Tatsächlich können die dynamischen Verhältnisse weitaus komplizierter sein, wenn Zwischenzustände keine Sattelpunkte, sondern metastabile Sattelmengen sind, in denen das System z. B. komplexe Schwingungen vollführt (Friston 1997, Rabinovich et al. 2008).

Zustandsraum eines solchen dynamischen Systems wird häufig mit der Menge der Aktivierungsmuster in einem neuronalen Netzwerk identifiziert, so dass die meisten konnektionistischen Modelle ebenfalls zur Klasse der dynamischen System-Modelle gezählt werden können.³

In der Psycho- und Neurolinguistik werden konnektionistische und dynamische Modelle seit Mitte der 1980er Jahre eingesetzt (für einen Überblick s. Elman et al. (1996) und den dazugehörigen Übungsband von Plunkett & Elman (1997)). So z. B. in den Spracherkennungsmodellen *TRACE* von McClelland & Elman (1986) und *NETtalk* von Sejnowski & Rosenberg (1987), in dem berühmten Modell von Rumelhart & McClelland (1986) zur Übergeneralisierung der englischen Vergangenheitsform und in den Wort-Erkennungsmodellen von Seidenberg & McClelland (1989) und Kawamoto (1993). Ausgehend von diesen Ansätzen modellierten Hinton & Shallice (1991) und Plaut (1996) neurolinguistische Symptome, wie Dyslexien, und deren Therapiemöglichkeiten in neuronalen Netzwerken durch lokale Läsionen und Wiedererlernen (s. dazu auch Plaut & Shallice (1993)), während die *Cell-Assembly*-Modelle von Wennekers et al. (2006) und Garagnani et al. (2007) Eigenschaften des mentalen Lexikons beschreiben und sogar in der Lage sind, ein neurophysiologisches Korrelat des Lexikonzugriffs, nämlich ein ereigniskorreliertes Potenzial (EKP, engl. *Event-Related Brain Potential*, ERP), die sogenannte *Mismatch Negativity* (MMN: Näätänen et al. (1993)) qualitativ vorherzusagen.⁴

In der Syntax wurden vornehmlich sogenannte *Simple-Recurrent-Networks* (SRN) zur Vorhersage von Buchstaben (Elman 1990) innerhalb eines Wortes oder von Worten und grammatischen Kategorien innerhalb eines Satzes (Elman 1991, 1995) eingesetzt. Diese Modelle erfreuen sich großer Beliebtheit in der computationalen Psycholinguistik. So erweiterten Tabor et al. (1997) und Tabor & Tanenhaus (1999) Elmans SRN-Architektur um einen dynamischen Wettbewerbs-Mechanismus (*Visitation Set Gravitation Model*), der zur Vorhersage von Lesezeiten dient, indem die Aktivierungsmuster in der verborgenen Schicht (*Hidden Layer*) des trainierten Netzwerks als Kraftzentren aufgefasst werden, die um den aktuellen Zustand konkurrieren und die Bahnkurve im Zustandsraum umzulenken versuchen. Für weitere SRN-Modelle siehe: Christiansen & Chater (1999), Lawrence et al. (2000), Farkas & Crocker (2008).⁵

³ Einen umfassenden Überblick auf Konnektionismus, Neuroinformatik und Systemneurobiologie liefern (in deutscher Übersetzung) Churchland & Sejnowski (1997).

⁴ Einen Überblick in diese Art der neurokognitiven Modellierung geben Braitenberg & Pulvermüller (1992).

⁵ S. auch die von Christiansen et al. (1999) und von beim Graben & Potthast (2009a) herausgegebenen Sonderbände der Zeitschriften *Cognitive Science* und *Cognitive Neurodynamics*.

Die oben genannten Ansätze haben allerdings den Nachteil, dass sie hauptsächlich von kontextfreien Beschreibungen natürlicher Sprache ausgehen (Carstensen et al. 2010, Hopcroft & Ullman 1994), während in der theoretischen Linguistik der Nachweis erbracht worden ist, dass natürliche Sprachen zur Klasse der sogenannten gemäßigt kontextsensitiven Sprachen gehören (Shieber 1985, Stabler 2004). Daher werden in jüngerer Zeit mächtigere Grammatik-Formalismen, wie z. B. *Baum-Adjunktions-Grammatiken* (*Tree Adjoining Grammars*, TAG; Joshi et al. (1975), Joshi & Schabes (1997)) oder *minimalistische Grammatiken* (*Minimalist Grammars*, MG; Stabler (1997), Stabler & Keenan (2003)) für die kognitive Modellierung eingesetzt (Vosse & Kempen 2000, 2009, Gerth & beim Graben 2009).

Ein weiteres Problem im Zusammenhang mit dem „eliminativen Konnektionismus“ (Blutner 2011, Churchland & Sejnowski 1997) ist, dass z. B. Elman (1991) und Tabor & Tanenhaus (1999) formale Grammatiken lediglich zur Erzeugung von Trainingsdaten verwenden. Ein SRN hat seine „Aufgabe gelernt“, wenn es die statistischen Regelmäßigkeiten in den Trainingsdaten durch neuronale Aktivierungsmuster im *Hidden Layer* „repräsentiert“ hat. Diese Repräsentationen sind überaus komplex und können mit statistischen Methoden, wie z. B. Clusteranalyse (Elman 1990) oder Hauptkomponentenanalyse (Elman 1991) allenfalls qualitativ untersucht werden. Ihre symbolische Beschreibung durch linguistische Berechnungen ist bisher kaum möglich (Gärtner & Jurish 2011).

Um indessen linguistische Berechnungen, wie z. B. Spracherkennung oder syntaktisches Parsing, im Rahmen der generativen Grammatik durch dynamische oder konnektionistische Systeme darstellen zu können (Schnelle 1996), haben Mizraji (1989) und Smolensky (1990) unabhängig voneinander einen mathematischen Formalismus vorgeschlagen, der inzwischen in die ICS- (*Integrated Connectionists/Symbolic*; Smolensky & Legendre (2006a, 2006b), Smolensky (2006)) und DCM- (*Dynamic Cognitive Modeling*; beim Graben & Potthast (2009b)) Architekturen eingebettet worden ist.

Mit diesem Übersichtsartikel beabsichtigen wir, die ICS/DCM-Verfahren der neurolinguistischen Modellierung an einem Beispiel aus der syntaktischen Sprachverarbeitung zu illustrieren. Dabei gehen wir von experimentellen Befunden zur Lizenzierung von *negativen Polaritäts-Elementen* (*Negative Polarity Items*, *NPI*) im Paradigma ereigniskorrelierter Gehirnpotenziale (Drenhaus et al. 2005, 2006, Drenhaus & beim Graben 2012) aus. Dazu werden zuerst die syntaktischen und semantischen Lizenzierungsbedingungen von NPI mithilfe minimalistischer Grammatiken beschrieben (Stabler 1997, Stabler & Keenan 2003). In einem zweiten Schritt werden die von einem minimalistischen Parser berechneten Zustandsbeschreibungen durch DCM auf Aktivierungsmuster eines neuronalen Netzwerks abgebildet (Gerth & beim Graben 2009). Dann werden die synaptischen Gewichte des Netzwerks so trainiert, dass das Netzwerk die korrekten NPI-

Lizenzierungsbedingungen realisiert. Mit den derart trainierten Gewichten können sowohl für die korrekten als auch für die nicht erfüllten Lizenzierungsbedingungen syntaktische Harmonie-Maße berechnet werden (Smolensky & Legendre 2006a, Smolensky 2006). Schließlich werden die Harmonie-Werte des dynamischen Modells mit den experimentell ermittelten EKP-Amplituden in einem einfachen statistischen Modell (Gries 2012) korreliert.

2 Methoden

Als computationale Neurolinguistik wollen wir hier die Vereinigung von Forschungsmethoden aus der Computerlinguistik und ihrer Teildisziplin, der computationellen Psycholinguistik, mit solchen aus der Neuroinformatik und der Neurolinguistik bezeichnen. Sie umfasst daher sowohl die formalen symbolischen Ansätze der Computerlinguistik als auch die formalen dynamischen und neuronalen Ansätze der computationellen Neurowissenschaft. Ihre Ziele sind, neuronale Korrelate linguistischer Berechnungen aufzufinden, wie dies z. B. von Hinton & Shallice (1991) und Plaut (1996) mit ihren neuronalen Dyslexie-Modellen, von Garagnani et al. (2007) mit der Modellierung der MMN im EKP, oder von Vosse & Kempen (2000) mit der Simulation syntaktischer Sprachverarbeitung von Aphasikern versucht worden ist.

2.1 Negative Polaritäts-Elemente

In einem EKP-Experiment zur Verarbeitung von NPI verwendeten Drenhaus et al. (2005) und Drenhaus et al. (2006) Aussagesätze der Form:

1. Kein Mann, der einen Bart hatte, war **jemals** glücklich.
2. * Ein Mann, der keinen Bart hatte, war **jemals** glücklich.
3. * Ein Mann, der einen Bart hatte, war **jemals** glücklich.

Ereigniskorrelierte Gehirnpotenziale wurden dabei auf dem *kritischen Wort*, dem NPI „jemals“ (fettgedruckt), gemessen.

Die Grammatikalität einer Polaritätskonstruktion ist zum einen davon abhängig, ob ein passender Lizenzierer vorhanden ist und zum anderen davon, ob dieser Lizenzierer strukturell zugänglich ist. Dies kann so interpretiert werden, dass sowohl semantische als auch syntaktische Lizenzierungsbedingungen erfüllt sein müssen. Für unsere einfache Modellierung identifizieren wir die semantische Lizenzierungsbedingung mit dem Vorhandensein einer Negation, während sich

das NPI nach der syntaktischen Lizenzierungsbedingung im Skopus des Lizenziers befinden muss.⁶

Demnach ist Satz 1 wohlgeformt, weil das NPI „jemals“ sowohl semantisch als auch syntaktisch lizenziert ist. Die semantische Lizenzierungsbedingung ist durch die Negation „kein“ gegeben. Nach der syntaktischen Lizenzierungsbedingung muss sich das NPI im Skopus des Lizenziers befinden, was für Satz 1 auch der Fall ist, weil „kein Mann“ das NPI „jemals“ c-kommandiert. Dagegen ist Satz 3 ungrammatisch, weil es überhaupt keinen Lizenzierer für das NPI gibt. In Satz 2 ist zwar durchaus ein Lizenzierer „kein“ vorhanden, welcher die semantische Lizenzierungsbedingung erfüllen könnte. Allerdings ist dieser in einem Relativsatz eingebettet, so dass die syntaktische Lizenzierungsbedingung nicht erfüllt ist.

2.1.1 Minimalistische Syntax

Um die Lizenzierungsbedingungen negativer Polaritäts-Elemente zu erfüllen, müssen semantische und syntaktische Merkmale über längere Reichweiten überprüft werden können. Dies ist mit den Mitteln kontextfreier Grammatiken nicht plausibel durchführbar, weshalb mächtigere Formalismen aus der Klasse der gemäßigt-kontextsensitiven Grammatiken (s. o.) zu verwenden sind. Einen für unsere Modellierung besonders geeigneten Ansatz im Rahmen der generativen Grammatik bieten Stablers minimalistische Grammatiken (MG), weil der syntaktische Strukturaufbau dort durch die Überprüfung von Merkmalen kontrolliert wird (Stabler 1997, Stabler & Keenan 2003, Stabler 2011a).

Eine MG besteht aus einem *Lexikon*, in dem Worte als Listen syntaktischer, phonetischer und semantischer Merkmale kodiert sind, und zwei strukturerzeugenden Funktionen, namens „merge“ und „move“. Syntaktische Merkmale im Lexikon sind die *Grundkategorien* d: Determinierer, v: Verb, n: Substantiv, a: Adjektiv, adv: Adverb, p: Präposition, t: Flexion und c: Komplementierer. Grundkategorien sind syntaktische Köpfe, die andere Kategorien als Komplemente oder Adjunkte selektieren (Frey & Gärtner 2002). Dies wird durch Selektor-Kategorien ausgedrückt, die mit den selektierten Kategorien durch „merge“ zu Phrasenstrukturbäumen verschmolzen werden. Selektoren können *starke* oder *schwache* Merkmale sein. Schwache Merkmale lösen lediglich Baumverschmelzung aus, während starke Merkmale zusätzlich Kopfbewegungen (normalerweise aus Komplementen heraus) hervorrufen (Stabler 1997). Außerdem unterscheidet man *Lizensierer* und *Lizenzierte*, die

⁶ Für eine ausführlichere Diskussion der Lizenzierungsbedingungen (Veridikalität, Monotonie) von NPI siehe Drenhaus & beim Graben (2012) und Verweise darin.

die Bewegung von Maximalprojektionen durch „move“ triggern. Auch Lizenzierer können stark oder schwach sein, um entweder overte oder kovertre Bewegungen zu beschreiben. In der Standard-Formulierung der MG ist die Reihenfolge von Grundkategorien, Selektoren, Lizenzierer und Lizenzierten in der lexikalischen Merkmalsliste wie folgt festgelegt: zwischen einer beliebige Anzahl von Selektoren darf höchstens ein Lizenzierer stehen. Dann folgt die Grundkategorie und daraufhin eine beliebigen Anzahl von Lizenzierten (Stabler 1997, Stabler & Keenan 2003).

Um die Lizenzierungsbedingungen von NPI zu kodifizieren, verallgemeinern wir hier den MG-Formalismus, indem nach einer beliebigen Anzahl von Lizenzierten auch wieder ein Lizenzierer stehen darf. Die Relativsätze 1–3 werden dabei durch die von Hale (2003, 2006) ausgearbeitete minimalistische Promotionsanalyse beschrieben. Die folgende Tabelle zeigt das minimalistische Lexikon für die Sätze 1–3.

der : : =n +F d -nom -wh	Mann : : n -f
ein : : =n d	ein : : =c d
Bart : : n	ε : : =t c
ε : : =t +WH c	k(neg) : : =d d -nom +npi
ε : : =d d -nom	k(neg) : : =d d -acc +npi
ε : : =d d -acc	sein : : =a =adv v
haben : : =d +ACC v	(pret) : : V= =d +NOM t
jemals : : adv -npi	glücklich : : a

Tabelle 1: Minimalistisches Lexikon für die Sätze 1–3

Tabelle 1 ist folgendermaßen zu interpretieren: Der erste Eintrag beschreibt den Artikel „der“, welcher die Grundkategorie d besitzt. Er selegiert als Komplement ein Substantiv, was durch den Selektor =n ausgedrückt wird. Mit dem starken Lizenzierer +F weist die resultierende DP einer anderen Maximalprojektion Fokus zu. Schließlich verlangt die DP Kasuszuweisung -nom und Wh-Bewegung -wh. Der zweite Eintrag ist das Substantiv „Mann“ (Grundkategorie n), dem vom vorigen Artikel Fokus zugewiesen wird: -f. An dritter und vierter Stelle im Lexikon stehen zwei unterschiedliche Einträge für den unbestimmten Artikel „ein“. Der dritte Eintrag ähnelt dem ersten: „ein“ selegiert ein Substantiv und ist selbst ein Artikel. Im vierten Eintrag selegiert „ein“ hingegen eine CP, also in unserem Beispiel den Relativsatz (und ist selbst ein Artikel). Der „Bart“ an fünfter Stelle ist ein einfaches Substantiv n. Die nächsten beiden Einträge stehen für phonetisch leere Lexeme: beide sind Komplementierer (Kategorie c), die eine IP selegieren (=t). Der siebte triggert allerdings auch eine overte Wh-Bewegung.

Mit dem achten Eintrag wird das semantische Merkmal der Negation dadurch ausgedrückt, dass *neg* in runden Klammern steht. Phonetisch ist die Verneinung durch das Präfix „k“ realisiert, das einer zu selektierenden DP angeheftet wird. Das Ergebnis ist wiederum eine DP, der in der weiteren Verarbeitung Nominativ zuzuweisen ist (-*nom*). Schließlich ist der Negationsoperator ein schwacher NPI-Lizensierer: +*npi*. Im Gegensatz zum achten drückt der phonetisch leere neunte Eintrag die Bejahung aus, indem einer selektierten DP das Bewegungsmerkmal -*nom* hinzugefügt wird. Diese Paarung spiegeln der zehnte und elfte Eintrag wieder, mit dem Unterschied, dass diesen DPs Akkusativ zugewiesen werden muss: -*acc*. Das zwölfte Lexem kodiert das Hilfsverb „sein“ als Verb *v*, das zuerst ein Adjektiv =*a* und danach ein Adverb =*adv* auswählt, um ein Prädikativum zu bilden. „Haben“, das 13. Lexem, ist ein Verb, das seinem Komplement, einer DP (=d), Akkusativ zuweist: +*acc*. An 14. Stelle steht ein phonetisch leerer Eintrag mit der Bedeutung Präteritum (*pret*), welcher zuerst ein Verb (*v*=) selektiert und durch Kopfbewegung mit Links-Adjunktion flektiert, danach eine DP als Subjekt (=d) auswählt, und ihr anschließend Nominativ zuweist (+*nom*), und der selbst von der Kategorie Flexion (*t*) ist. Es folgt das NPI „jemals“, das ein Adverb ist (*adv*) und lizenziert werden muss (-*npi*). An letzter Stelle steht schließlich das Adjektiv „glücklich“ (*a*).

Diese Lexeme werden durch die strukturbildenden Funktionen zu Phrasenstrukturbäumen zusammengesetzt. Ein minimalistischer Phrasenstrukturbaum ist entweder einfach oder komplex. Einfache Bäume sind Einträge des Lexikons; komplexe Bäume entstehen durch „merge“. Ihre Blätter sind Merkmalslisten, während Zwischenknoten mit den *Projektionsanzeigern* „<“ und „>“ etikettiert sind. Jeder Baum hat einen eindeutig bestimmten *Kopf*: einfache Bäume sind selbst Köpfe, während der Kopf eines komplexen Baums ermittelt wird, indem man dem durch die Projektionsanzeiger gewiesenen Pfad von der Wurzel zu einem Blatt folgt. Andererseits ist jedes Blatt auch Kopf eines Teilbaumes, so dass dieser eine *Maximalprojektion* ist. Als das Merkmal eines Baumes vereinbart man das erste Merkmal in der Merkmalsliste des Kopfes. Die partielle Funktion

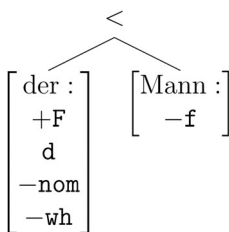


Abbildung 1: Ergebnis der merge-Funktion angewendet auf die Lexeme „der“ und „Mann“ aus dem minimalistischen Lexikon Tab. 1.

„merge“ bildet zwei Bäume, von denen einer ein Selektor-Merkmal und der andere die dazugehörige Grundkategorie hat, auf den verschmolzenen Baum ab, wobei der Projektionsanzeiger auf den vormaligen Selektor verweist, beide Merkmale aber nach der Verschmelzung getilgt werden. Abbildung 1 zeigt das Ergebnis von merge (der, Mann): Die Merkmale der einfachen Bäume, $=n$ von „der“ und n von „Mann“ wurden nach der Überprüfung und Verschmelzung getilgt. Der Kopf des resultierenden komplexen Baumes ist „der“, weil der Pfeil „<“ von der Wurzel auf diesen Knoten weist, so dass der gesamte Baum eine DP darstellt.

Nach der merge-Operation, liegt der Baum in Abb. 1 in der Domäne der move-Funktion, weil der gesamte Baum das starke Merkmal $+F$ und genau eine Maximalprojektion, nämlich das Blatt „Mann“ das Merkmal $-f$ trägt.⁷ Durch „move“ wird diese Maximalprojektion overt in eine Spezifizierer-Position bewegt, während an der ursprünglichen Stelle eine leere Kategorie λ eingefügt wird. Die Indizes i sollen die Bewegung veranschaulichen; sie gehören allerdings nicht zum Formalismus der MG.

Im nächsten Schritt würde der Baum aus Abb. 2 mit Merkmal \bar{d} mit einem Baum mit Merkmal $=\bar{d}$ verschmolzen werden, dies ist in unserem Beispiel die IP „einen Bart hatte“, so dass nach weiteren Operationen der Relativsatz „Mann der einen Bart hatte“ (Merkmal \bar{c}) entsteht.⁸

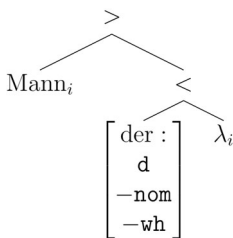


Abbildung 2: Ergebnis der move-Funktion angewendet auf den Baum aus Abb. 1.

⁷ Die Bedingung, dass genau eine Maximalprojektion das zum Lizenzierer passende Lizenzierte besitzt, wird als *Shortest-Move-Condition (SMC)* bezeichnet. Diese Lokalisierungsbedingung spielt eine wichtige Rolle für die Komplexität minimalistischer Grammatiken (Gärtner & Michaelis 2005).

⁸ Da der minimalistische Baum-Formalismus in der Darstellung recht aufwändig ist, haben Harkema (2001) und Stabler & Keenan (2003) einen vereinfachten Ketten-Formalismus eingeführt, den wir im Anhang verwenden, um die minimalistischen Ableitungen der Beispielsätze 1–3 anzugeben (dabei kommt es im Vergleich zum Baum-Formalismus, der für die MG aus Tab. 1 korrekte Ableitungen liefert, zu Unterschieden in der Wortordnung).

Unsere Modifikation der MG macht eine weitere move-Operation erforderlich, die wir in Abb. 3 skizzieren. Ziemlich am Ende der Ableitung von Satz 1 entsteht ein Baum der Form Abb. 3(a).

Der Kopf dieses Baumes ist „war“, so dass sein Merkmal $+NOM$ ist. Andererseits gibt es die Maximalprojektion „kein Mann der einen Bart hatte“ mit Merkmal $-nom$, die demnach zu bewegen ist. Dabei stört indes das nachfolgende Merkmal $+npi$, das in einer nichtkanonischen Position steht. Da es sich hierbei um einen Lizensierer handelt, der nur aus einer Kopfposition heraus Bewegung auszulösen vermag, muss die move-Funktion entsprechend modifiziert werden: Wir schlagen daher vor, die Bewegung wie üblich auf das phonetische und semantische

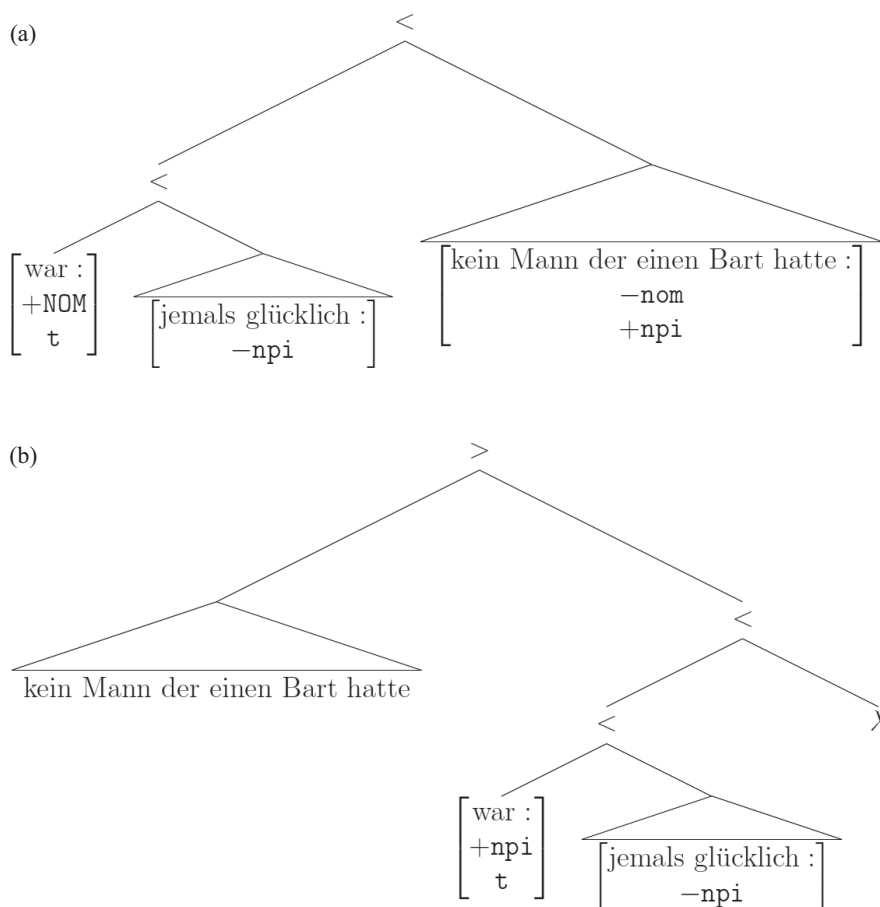


Abbildung 3: Nichtkanonische Bewegung.

Material anzuwenden, jedoch die syntaktischen Merkmale an die des Kopfes „war“ zu präfigieren. Das Ergebnis dieser nichtkanonischen Bewegung zeigt Abb. 3(b).

Der Baum in Abb. 3(b) erlaubt eine abschließende koverte Bewegung, durch welche die NPI-Lizensierung in unserem Modell durchgeführt wird.⁹ Das Ergebnis ist die IP „kein Mann der einen Bart hatte war jemals glücklich“, die in einem letzten Schritt mit dem phonetisch leeren Komplementierer *c* verschmolzen wird, um eine CP zu erhalten. Damit ist die syntaktische Analyse des wohlgeformten Satzes 1 abgeschlossen.

Die Sätze 1–3 werden mit einem Ketten-basierten minimalistischen Parser (s. Anhang) bis zum Abbruch verarbeitet. Die Zustandsbeschreibungen des Parsers bilden die Kollektionen von Ausdrücken, die durch die Anwendung der Funktionen „merge“ und „move“ auf eine anfängliche Enumeration des Lexikons hervorgehen.

2.1.2 Ereigniskorrelierte Gehirnpotenziale

Sätze der Form 1–3 dienten als Stimulus-Material für die EKP-Studie von Drenhaus et al. (2005) und Drenhaus et al. (2006). Für unsere Globalanalyse wählen wir als Epochenlänge das Zeitfenster des gesamten Satzes von –3700 ms (Satzanfang bezogen auf die Darbietung des NPI zur Zeit 0) bis 1200 ms (Satzende).¹⁰

Da die korrekte Bedingung 1 die Grundlinie definieren soll, subtrahieren wir das *grand average* EKP für 1 von jedem einzelnen Versuchsperson-EKP für jede Bedingung. Diese Differenzen werden dann quadriert und über den Zeitverlauf des gesamten Satzes gemittelt, um die EKP-Leistung je Versuchsperson und Bedingung zu bestimmen.

2.2 Neuronale Netzwerkmodelle

In der computationalen Neurowissenschaft gibt es im Wesentlichen zwei Hauptströmungen der aktuellen Forschung: In der einen versucht man der schon unge-

⁹ Da mehrere NPI auch von einer Negation lizensiert werden können, müsste der Formalismus noch um einen „NPI-Clustering-Mechanismus“ erweitert werden, wie ihn Gärtner & Michaelis (2010) für das *Wh-Clustering* vorgeschlagen haben, um die SMC nicht zu gefährden.

¹⁰ Die EKP-Analyse wurde unter Verwendung der freien MATLAB-Toolbox EEGLAB (Delorme & Makeig 2004) durchgeführt. Dabei konnte eine Versuchsperson (#5) nicht importiert werden, so dass wir anstatt der von Drenhaus et al. (2005) und Drenhaus et al. (2006) berichteten 16 Versuchspersonen nur 15 auswerten konnten.

heuren biologischen Komplexität einzelner Nervenzellen gerecht zu werden, während in der anderen die nicht minder bedrückende Komplexität neuronaler Netzwerke im Vordergrund steht. Die Informationsverarbeitungseigenschaften eines einzelnen Neurons erklären sich aus der Biophysik der Zellmembranen, in die unterschiedlichste Rezeptoren und Ionenkanäle eingebettet sind. Bereits deren Modellierung erfordert Systeme gekoppelter nichtlinearer Differentialgleichungen, die lediglich für kleinere Verschaltungen durch Computersimulation lösbar sind.¹¹

Auf der anderen Seite lassen sich komplexe neuronale Netzwerke, wie sie auch für die konnektionistische Modellierung in der computationellen Neurolinguistik relevant sind, nur durch erhebliche Abstraktion von der biologisch-physiologischen Beschreibung gewinnen (Churchland & Sejnowski 1997). Dieser Zugang soll im Folgenden etwas ausgeführt werden.

2.2.1 Allgemeine Übersicht

Bei der konnektionistischen Modellierung werden von den Eigenschaften realer biologischer Neurone lediglich zwei als wesentlich berücksichtigt (Kandel et al. 1996): 1. Das *Prinzip der gewichteten Summierung*; und 2. das *Alles-oder-Nichts-Gesetz*, die wir im Folgenden veranschaulichen wollen.

Abbildung 4 zeigt ein sehr einfaches neuronales Netzwerk bestehend aus drei Modell-Neuronen: 1, 2, 3, die wir im Folgenden als *Einheiten* bezeichnen wollen. Eine Einheit ist durch eine Zustandsgröße, ihre *AKTIVITÄT*, gekennzeichnet, die entweder gleich Null ist: die Einheit ist inaktiv, oder gleich Eins ist, wenn die Einheit „feuert“.

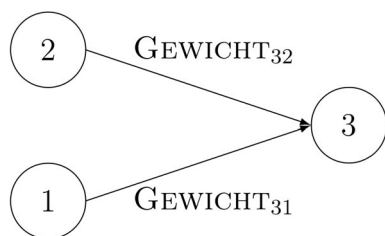


Abbildung 4: Einfaches neuronales Netzwerk.

¹¹ Einen Überblick über diesen Forschungszweig bietet das von beim Graben et al. (2008b) herausgegebene Buch *Lectures in Supercomputational Neuroscience*, insbesondere Kapitel 1 (beim Graben 2008).

Einheiten 1 und 2 leiten in Vorwärtsrichtung Signale zur Einheit 3, weswegen man die hier vorliegende Architektur als *feedforward Topologie* bezeichnet. Darüber hinaus sind die Verbindungen von 1 nach 3 und von 2 nach 3 gewichtet, so dass GEWICHT_{31} das Gewicht der Verbindung von 1 nach 3 und GEWICHT_{32} das Gewicht der Verbindung von 2 nach 3 bezeichnet.¹²

Mit dieser Notation können wir eine wichtige Kenngröße der Einheit 3, ihren NETINPUT als gewichtete Summe berechnen:

$$\text{NETINPUT}_3 = \text{GEWICHT}_{31} \times \text{AKTIVITÄT}_1 + \text{GEWICHT}_{32} \times \text{AKTIVITÄT}_2, \quad (1)$$

wobei die Gewichte positiv (*exzitatorisch*) oder negativ (*inhibitorisch*) sein können.

Der NETINPUT einer Einheit kann in grober Näherung mit dem *Membranpotenzial* eines biologischen Neurons identifiziert werden, wodurch eine Zelle zum Elektroenzephalogramm (EEG) und damit zum ereigniskorrelierten Potenzial (EKP) beiträgt (Liley et al. 2002, beim Graben & Kurths 2008). Der NETINPUT von Einheit 3 entscheidet nun darüber, ob diese selbst feuert oder inaktiv bleibt, indem er mit einem vorgegebenen Schwellenwert, der Aktivierungsschwelle, SCHWELLE_3 , von Einheit 3 verglichen wird:

$$\text{AKTIVITÄT}_3 = \begin{cases} 0 & \text{für } \text{NETINPUT}_3 \leq \text{SCHWELLE}_3 \\ 1 & \text{für } \text{NETINPUT}_3 > \text{SCHWELLE}_3 \end{cases} \quad (2)$$

Gleichung (2) formalisiert das Alles-oder-Nichts-Gesetz der Neurophysiologie (Kandel et al. 1996). Die derart beschriebenen Einheiten werden als McCulloch-Pitts-Einheiten bezeichnet. Wie McCulloch & Pitts (1943) gezeigt hatten, können Netzwerke von solchen Einheiten zur Konstruktion von Digitalcomputern verwendet werden, wie ein einfaches Zahlenbeispiel zeigt. Setzen wir für das obige Netzwerk die Werte

$$\text{GEWICHT}_{31} = 0,25$$

$$\text{GEWICHT}_{32} = 0,25$$

$$\text{SCHWELLE}_3 = 0,4$$

ein, erhalten wir als Wertetabelle:

¹² Die Reihenfolge der Indizes ist eine Konvention, die durch den Matrizenformalismus der linearen Algebra nahegelegt wird.

Aktivität ₁	Aktivität ₂	NetInput ₃	Aktivität ₃
0	0	$0,25 \cdot 0 + 0,25 \cdot 0 = 0$	0
1	0	$0,25 \cdot 1 + 0,25 \cdot 0 = 0,25$	0
0	1	$0,25 \cdot 0 + 0,25 \cdot 1 = 0,25$	0
1	1	$0,25 \cdot 1 + 0,25 \cdot 1 = 0,5$	1

Tabelle 2: Wertetabelle für NetInput und Aktivität der logischen Und-Funktion

Dies ist gerade die Wahrheitstafel der aussagenlogischen Und-Funktion. Prinzipiell wäre es also durchaus möglich, ein Gehirn aus einfachen McCulloch-Pitts-Einheiten als Digitalcomputer zusammenzusetzen, wie die in der Einleitung erwähnte „Computermetapher des Geistes“ nahe legt. Dem steht indes nicht nur die Komplexität biologischer Nervenzellen entgegen, sondern vor allem das allgegenwärtige thermische Rauschen in natürlichen Systemen. Daher werden in der Neuroinformatik die Zustandsgrößen neuronaler Netzwerke als (in der Regel) normal verteilte Zufallsvariablen betrachtet. Ersetzt man dann die bisherigen Größen Aktivität und NetInput durch ihre jeweiligen Mittelwerte, bestimmt das Alles-oder-Nichts-Gesetz nunmehr die Wahrscheinlichkeit, dass der NetInput einer Einheit ihre Schwelle übersteigt. Diese ist, bei einer Normalverteilung, durch die sigmoidale Verteilungsfunktion bestimmt, die in Computersimulationen näherungsweise durch die ähnlich geformte logistische Funktion

$$f(x) = \frac{1}{1 + e^{-\beta x}} \tag{3}$$

ersetzt wird (vgl. Gries (2012)). Der zusätzliche Parameter β wird als *Verstärkung* bezeichnet und bestimmt die Steigung der Aktivierungsfunktion in der Nähe der Null.

In der stochastischen Beschreibung ist daher Gl. (2) durch

$$\text{Aktivität}_3 = f(\text{NetInput}_3 - \text{Schwelle}_3) \tag{4}$$

zu ersetzen. Die Gleichungen (1) und (4) können geradewegs auf neuronale Netzwerke aus beliebig vielen Einheiten und von beliebiger Verschaltung (Topologie) verallgemeinert werden. Dadurch wird (1) zu einem *linearen Gleichungssystem*, das kompakter als *Matrizen-Gleichung*

$$\text{NetInput} = \text{Gewicht} \cdot \text{Aktivität} \tag{5}$$

mit *Gewichtsmatrix* GEWICHT und Zustandsvektoren AKTIVITÄT und NETINPUT notiert werden kann. Für *rekurrente Netzwerke*, bei denen Signale zwischen Einheiten rückgekoppelt werden, kombinieren sich Gleichungen (4) und (5) zu einer Bewegungsgleichung für ein dynamisches System:

$$\text{AKTIVITÄT}(\text{ZEIT}+1) = f(\text{GEWICHT} \cdot \text{AKTIVITÄT}(\text{ZEIT}) - \text{SCHWELLE}) \quad (6)$$

wodurch der Aktivierungsvektor des Netzwerks von einem Zeitpunkt zum nachfolgenden aktualisiert wird.

Im obigen Beispiel für die logische Und-Funktion konnten wir Gewichte und Schwellen von Hand einstellen. Dies ist bei komplexen neuronalen Netzwerken aus sehr vielen Einheiten natürlich nicht möglich. Dafür ist jedoch häufig die Abfolge von Zustandsvektoren, also eine *Bahnkurve* AKTIVITÄT(ZEIT) für eine Folge von Zeitpunkten bekannt. Diese dient dann als Trainingsmenge für die Bestimmung der System-Parameter GEWICHT und SCHWELLE. Mathematisch läuft das Training neuronaler Netzwerke auf die Umkehrung von Gl. (6) hinaus. Dafür haben Potthast & beim Graben (2009) und beim Graben & Potthast (2009b) ein Lösungsverfahren entwickelt, das die bekannte *Hebbsche Lernregel*, nach der eine Synapse zwischen zwei Neuronen genau dann verstärkt wird, wenn beide Zellen gleichzeitig aktiviert sind (Hebb 1949), verallgemeinert. Dieses Tikhonov-Hebb-Verfahren steht im Mittelpunkt von DCM (*Dynamic Cognitive Modeling*) und wird im folgenden Abschnitt zum Training der NPI-Verarbeitung verwendet.

2.2.2 Modell-Implementation

In diesem Abschnitt soll ein neuronales Netzwerk mit den Ausgaben des minimalistischen Parsers trainiert werden. Als Architektur wird ein vollständig rekurrentes Netzwerk verwendet, bei dem grundsätzlich jede Einheit mit jeder anderen verbunden ist, auch wenn am Ende des Trainings Verbindungen dadurch verloren gehen können, dass ihre Gewichte zu Null werden. Solche Netzwerke werden als *Hopfield-Netze* bezeichnet (Hopfield 1982, Churchland & Sejnowski 1997), die zur *Muster-Erkennung* eingesetzt werden. Hopfield-Netze sind dynamische Systeme mit Attraktoren, deren Zustände nach einer transienten Berechnungsphase in einen *Fixpunkt-Attraktor* hineinlaufen, der dem Endergebnis einer Berechnung entspricht. Wie Siegelmann & Sontag (1995) gezeigt haben, können Turing-Maschinen durch Hopfield-Netze realisiert werden, so dass diese Architektur in paradigmatischer Weise für die Modellierung in der computationellen Neurolinguistik geeignet ist.

Die Aktivierungs-Dynamik in einem Hopfield-Netz kann als Bewegung eines Punktes in einer Energie- bzw. Harmonie-Landschaft interpretiert werden, wobei

Harmonie die Konsistenz zwischen Aktivierungen und synaptischen Verbindungen angibt (Smolensky & Legendre 2006a, Smolensky 2006)). Als Formel:

HARMONIE = AKTIVITÄT · (GEWICHT · AKTIVITÄT) (7)

D. h. Harmonie ist das Skalarprodukt zwischen dem Aktivierungsvektor und dem Matrizenprodukt von Gewichtsmatrix und Aktivierungsvektor.

2.2.2.1 Fraktale Tensorprodukt-Darstellung

In einem ersten Schritt werden die Satzbeispiele 1–3 des EKP-Experiments von Drenhaus et al. (2005) und Drenhaus et al. (2006) durch einen minimalistischen Parser in zeitliche Abfolgen von Zustandsbeschreibungen übersetzt. Jede dieser Zustandsbeschreibung ist eine Tabelle von MG-Ketten, die im zweiten Schritt mit der sogenannten *fraktalen Tensorprodukt-Darstellung* (Siegelmann & Sontag 1995, Tabor 2000, beim Graben & Potthast 2009b) auf einen Aktivierungsvektor für ein Hopfield-Netz abgebildet wird. Ausgangspunkt dieser Darstellung ist eine *Gödelkodierung* der Typen („:“ = lexikalisch, „:“ = abgeleitet) und sämtlicher syntaktischer Merkmale in Tab. 1 durch natürliche Zahlen von Null bis 25, so dass die Anzahl dieser Symbole gleich 26 ist. Tabelle 3 zeigt die von uns gewählte Kodierung.

Symbol	Gödelkode	Symbol	Gödelkode
::	0	:	1
adv	2	=adv	3
a	4	=a	5
n	6	=n	7
d	8	=d	9
v	10	V=	11
t	12	=t	13
c	14	=c	15
+NOM	16	-nom	17
+ACC	18	-acc	19
+F	20	-f	21
+WH	22	-wh	23
+npi	24	-npi	25

Tabelle 3: Gödelkodierung des Lexikons aus Tab. 1.

Mithilfe dieser Gödelkodierung werden die minimalistischen Ketten aus Tab. 1 auf Bruchzahlen abgebildet, indem wir ein Stellenwert-System zur Basis 26 verwenden. Beispielsweise ergibt die Kodierung des Lexikoneintrages für „haben“,¹³

haben::=d +ACC v

nach Tab. 3, folgende rationale Zahl:

$$\text{GÖDEL(haben)} = 0 \times 26^{-1} + 9 \times 26^{-2} + 18 \times 26^{-3} + 10 \times 26^{-4} = 0,0144. \quad (8)$$

Die Gödelcodes der Ketten bilden die sogenannten (einfachen) *Füller* für die auf Mizraji (1989) und Smolensky (1990) zurückgehende Tensorprodukt-Darstellung (s. auch Smolensky & Legendre (2006a, 2006b), Smolensky (2006), beim Graben & Potthast (2009b)). Ein *minimalistischer Ausdruck* ist eine Liste von Ketten (Stabler & Keenan 2003), für die wir die *Rollenpositionen* 1. Element, 2. Element, ..., *n*-tes Element als kanonische Basisvektoren in einem *n*-dimensionalen Vektorraum darstellen (z. B. für *n*=2):

$$\text{POSITION1} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \text{POSITION2} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \quad (9)$$

Die Tensorprodukt-Darstellungen (TPR) von Ausdrücken ergeben sich dann wie folgt. Dem einfachen Strukturbaum aus Abb. 1 entspricht der Ausdruck

der:+F d -nom -wh, Mann:-f

Dieser wird kodiert, indem die Gödelnummer des ersten Eintrags als Füller an die 1. Rollenposition *gebunden* wird. Entsprechend wird der Gödelcode des zweiten Eintrags an die 2. Rollenposition gebunden. Formal wird diese *Füller/Rollen-Bindung* (Smolensky & Legendre 2006a, 2006b, Smolensky 2006) durch *Tensorprodukte* („ \otimes “) ausgedrückt, die aber in unserem einfachen Beispiel durch die Multiplikation von Zahlenwerten mit Vektoren realisiert sind:

¹³ Da uns hier lediglich die syntaktische Verarbeitung interessiert, werden phonetische und semantische Merkmale der Grammatik bei der Kodierung nicht berücksichtigt. Dadurch reduziert sich auch die Kopfbewegung auf reines „merge“.

$$\begin{aligned}
 \text{TPR}(\text{AUSDRUCK}) = \\
 \text{TPR}(1. \text{KETTE}) \otimes \text{POSITION1} + \text{TPR}(2. \text{KETTE}) \otimes \text{POSITION2} = \\
 (1 \times 26^{-1} + 20 \times 26^{-2} + 8 \times 26^{-3} + 17 \times 26^{-4} + 23 \times 26^{-5}) \times \begin{pmatrix} 1 \\ 0 \end{pmatrix} + (1 \times 26^{-1} + 21 \times 26^{-2}) \times \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \\
 \begin{pmatrix} 0,0685 \\ 0,0695 \end{pmatrix}. \quad (10)
 \end{aligned}$$

Die Zustandsbeschreibungen des minimalistischen Parsers werden dargestellt, indem man die Tensorprodukt-Darstellungen der Ausdrücke auf einem Speicherband berechnet und diese als komplexe Füller an jeweilige Rollenpositionen für diesen Speicher anbindet. Im Gegensatz zu oben müssen diese Füller/Rollen-Bindungen tatsächlich als Tensorprodukte von Vektoren mit Vektoren ausgeführt werden.¹⁴ Durch die Kombination von Gödelzahlen und Vektordarstellungen ergibt sich eine fraktale Struktur im Darstellungsraum (Siegelmann & Sontag 1995, Tabor 2000, beim Graben & Potthast 2009b). Ein minimalistischer Parse wird dadurch auf eine Vektorfolge, d. h. auf eine Bahnkurve im Darstellungsraum abgebildet.

2.2.2.2 Netzwerk-Training

Die durch fraktale Tensorprodukt-Darstellungen gewonnenen Bahnkurven dynamischer minimalistischer Parser sind Abfolgen überaus spärlicher und hochdimensionaler Aktivierungsvektoren,¹⁵ die eine Dimensionsreduzierung erforderlich machen, bevor die nächsten Implementationsschritte vollzogen werden können. Dazu verwenden wir die Hauptkomponentenanalyse (*PCA: Principal Component Analysis*; s. Elman (1991), beim Graben et al. (2008a)), die auf die standardisierte Verteilung des korrekten Parses von Satz 1 angewandt, sechs Richtungen größter Varianz ergibt. In diesen sechs-dimensionalen Unterraum projizieren wir auch die abgebrochenen Parses der Sätze 2 und 3, um ein Hopfield-Netz aus sechs vollständig miteinander verbundenen Einheiten zu trainieren.¹⁶ Trainiert wird lediglich die Bahnkurve für den korrekten Parse mit

¹⁴ Durch die rekursive und prinzipiell unendliche Überlagerung solcher Tensorprodukte entsteht der sogenannte *Fock-Raum* (Smolensky & Legendre 2006a, beim Graben et al. 2008a), der normalerweise in der Elementarteilchen-Physik verwendet wird.

¹⁵ Für unseren Ketten-basierten Parser werden 45 Raumdimensionen benötigt.

¹⁶ Um die Anzahl der relevanten Hauptkomponenten und dadurch die Größe des Hopfield-Netzes zu bestimmen, haben wir eine einfache Heuristik verwendet, bei der der Korrelationskoeffizient des statistischen Modells (s. u.) in Abhängigkeit der PCA-Projektion betrachtet wurde. Für einen sechs-dimensionalen Zustandsraum wurde dieser maximal.

dem von Potthast & beim Graben (2009) und beim Graben & Potthast (2009b) entwickelten Tikhonov-Hebb-Algorithmus.¹⁷ Dieser Schritt liefert als Ergebnis die synaptische Gewichtsmatrix *GEWICHT* für die Bewegungsgleichung (6) des korrekten Parses.

2.2.2.3 Harmonie-Berechnung

Im letzten Schritt der Modell-Implementation werden mithilfe der Gewichtsmatrix *GEWICHT* die Harmonien Gl. (7) für alle drei Parses 1–3 berechnet. Da die Harmonie am Ende des korrekten Parses 1 Null ergeben soll, subtrahieren wir diesen Wert von allen anderen Harmoniewerten, um die Harmonieskala zu eichen. Schließlich werden die Harmoniewerte für jeden einzelnen Parse aufaddiert, um die global kumulierte Harmonie *KUMHARMONIE* je Parse zu erhalten. Diese drei Werte gehen als Prädiktor in unser einfaches statistisches Modell ein.

2.2.2.4 Statistische Modellierung

Für die statistische Modellierung (Gries 2012) beschränken wir uns aus didaktischen Gründen auf ein einfaches Regressionsmodell mit der oben gewonnenen kumulierten Harmonie *KUMHARMONIE* als Prädiktor und der globalen EKP-Leistung *PzEKP* auf der Hinterhaupt-Elektrode *Pz* je Versuchsperson als abhängiger Variable. Als Modell setzen wir ein Potenzgesetz

$$PzEKP = A \times KUMHARMONIE^B \quad (11)$$

mit Parametern *A* und *B* an.

3 Ergebnisse

Die Ergebnisse der EKP-Studie von Drenhaus et al. (2005) und Drenhaus et al. (2006) werden ausführlich in Drenhaus & beim Graben (2012) dargestellt. Abbildung 3 zeigt dort die EKP-Mittelwerte für die beiden ungrammatischen Bedingungen 2 und 3 im Vergleich zur Kontrollbedingung 1. Bezogen auf das kritische Wort, das NPI „jemals“, wurde ein N400/P600-Muster in den Bedingungen 2 und 3 bezogen auf die Kontrollbedingung 1 beobachtet.

Das Ergebnis unseres statistischen Modells zeigt Abb. 5. Hier wurden auf der Abszisse die drei kumulierten Harmoniewerte aus dem neuronalen Netzwerk-

¹⁷ Trainingsparameter: Tikhonov-Regularisierungs-Parameter $\alpha = 0,1$; Verstärkung $\beta = 10$; Schwellenwerte *SCHWELLE* = 0,3 (einheitlich für alle sechs Einheiten).

modell von Abschnitt 2.2.2 als Prädiktor abgetragen. Auf der Ordinate plotten wir die EKP-Leistung auf der Elektrode Pz je Versuchsperson und Bedingung als abhängiger Variable. Dabei bedeuten „◇“: Bedingung 1, „○“: Bedingung 2 und „×“: Bedingung 3.

Durch „□“ werden für alle drei Bedingungen die über Versuchspersonen gemittelten EKP-Leistungen, also *grand averages* angezeigt. Diese sind durch die Regressionskurve 1. Art (gestrichelt) miteinander verbunden. Das statistische Modell Gl. (11) liefert die durchgezogene Linie als Regressionskurve 2. Art mit den Parametern: $A = 4,97$ und $B = 0,11$ ($r^2 = 0,031$; $F = 1,36$; $p = 0,25$). Konfidenzintervalle zu diesem 75 %-Niveau sind $\Delta_A = [4,21, 5,85]$ und $\Delta_B = [0,08, 0,14]$.

Wir finden trotz der geringen Korrelation von 0,031, die zudem mit einer Fehlerwahrscheinlichkeit von 25 % behaftet ist, eine monotone Abhängigkeit der EKP-Leistung von der Verarbeitungs-Harmonie. Damit bestätigt unser konnektio-

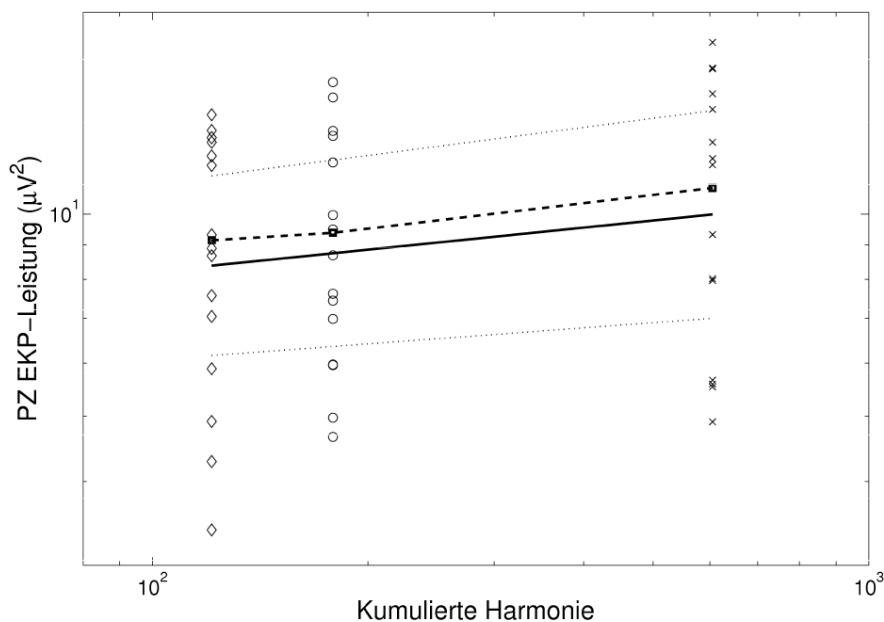


Abbildung 5: EKP-Leistung je Versuchsperson ($n = 15$) in Abhängigkeit der kumulierten Netzwerk-Harmonie. „◇“: Bedingung 1, „○“: Bedingung 2, „×“: Bedingung 3. Die durch die gestrichelte Linie (Regressionskurve 1. Art) verbundenen Quadrate („□“) bezeichnen die Leistungen gemittelt über alle Versuchspersonen. Die durchgezogene Linie ist die Regressionskurve 2. Art. Gepunktet ist das 75 % Konfidenzintervall der Regression.

nistisches Modell den vorliegenden Intrusions-Effekt, den Vasisht et al. (2008) mittels einer kognitiven Architektur vorhergesagt haben.¹⁸

4 Diskussion

In diesem Beitrag haben wir an einem Beispiel aus der Satzverarbeitung negativer Polaritäts-Elemente (NPI) im Paradigma ereigniskorrelierter Hirnpotenziale (EKP: Drenhaus & beim Graben (2012)) illustriert, welche einzelnen Schritte von einer symbolisch-linguistischen Beschreibung des Stimulusmaterials zu einem neuronal-dynamischen Modell führen. Um die Lizenzierungsbedingungen von NPI zu beschreiben, sind wir von einer minimalistischen Grammatik (MG: Stabler (1997), Stabler & Keenan (2003)) ausgegangen, die geringfügig zu modifizieren war. Durch die strukturerzeugenden Funktionen „merge“ und „move“ baut ein *bottom-up* Parser Kollektionen von Ausdrücken auf, die wir mithilfe einer fraktalen Tensorprodukt-Darstellung und anschließender Dimensionsreduktion auf Aktivierungsvektoren in einem Hopfield-Netz abgebildet haben (beim Graben & Pott-hast 2009b, Gerth & beim Graben 2009). Trainiert man die synaptischen Gewichte des Netzwerks mit dem korrekten Parse für NPI-Konstruktionen, lassen sich sowohl für diesen wie auch für die ungrammatischen Bedingungen Harmoniewerte bestimmen (Smolensky & Legendre 2006a, 2006b, Smolensky 2006), die in einem statistischen Modell (Gries 2012) mit den beobachteten EKP korreliert werden konnten.

Aus didaktischen Gründen haben wir uns in der vorliegenden Arbeit darauf beschränkt, eine Globalanalyse der EKP mit kumulierten Harmonien, beide über den gesamten Satzverlauf gerechnet, auf einer einzelnen Ableit-Elektrode (Pz: parietozentral) zu berichten. Dadurch reduziert sich unser statistisches Modell auf eine einfache Regression mit Harmonie als Prädiktor und EKP-Leistung als abhängiger Variable. Dieser einfache Ansatz kann in verschiedene Richtungen erweitert und verallgemeinert werden.

1. Der verwendete minimalistische *bottom-up* Parser arbeitet nicht inkrementell, so dass sich mit seiner Hilfe keine lokalen Vorhersagen über die Satzverarbeitung treffen lassen. In jüngster Zeit konnten Mainguy (2010) und Stabler (2011b), aufbauend auf theoretischen Untersuchungen von Harkema (2001), inkrementelle *top-down* Parser für MG entwickeln, die sich mit denselben Metho-

¹⁸ Das Modell von Vasisht et al. (2008) implementiert den Intrusions-Effekt durch partielles *Cue-Matching* beim Wiederabruf der NPI-Lizenzierungsbedingen. Unser neurokognitives Modell kommt ohne dergleichen Hypothesen aus.

den von DCM (beim Graben & Potthast 2009b), also Füller/Rollen-Bindungen und Tensorprodukt-Darstellungen (Smolensky & Legendre 2006a, 2006b, Smolensky 2006), neuronal implementieren lassen. 2. Anstelle von kumulierten Harmonien lassen sich für einen inkrementellen Parser Harmonie-Zeitreihen bestimmen, die sich mit EKP-Zeitreihen korrelieren lassen. 3. Das Modell lässt sich auf mehrere Prädiktoren und deren Interaktionen verallgemeinern, z. B. indem man statt Harmonien direkt die Richtungen größter Varianz im neuronalen Aktivierungsraum, also Hauptkomponenten verwendet. 4. Künftige Studien auf diesem Gebiet der computationellen Neurolinguistik sollten multivariate statistische Modelle verwenden, bei der als abhängige Variablen nicht lediglich ein EEG-Kanal sondern mehrere (oder gleich alle) verwendet werden. Dadurch würden auch topographische Unterschiede kortikaler Aktivität berücksichtigt.

Danksagung: Wir danken den Herausgebern der *Zeitschrift für Germanistische Linguistik* für die freundliche Einladung zu diesem Beitrag und Sabrina Gerth, Andreas Farner, Stefan Frisch, Greg Kobele und Hans-Martin Gärtner für hilfreiche Unterstützung. Diese Arbeit wurde durch ein Heisenberg-Stipendium (GR 3711/1-1) der Deutschen Forschungsgemeinschaft gefördert.

Anhang

In diesem Anhang präsentieren wir die minimalistischen Ableitung des korrekten Satzes 1 und die scheiternden Ableitungen der ungrammatischen Sätze 2 und 3 im Ketten-Formalismus (Stabler & Keenan 2003), der auch unserem minimalistischen *bottom-up* Parser zugrunde liegt. Eine *minimalistische Kette* ist eine Abfolge von lexikalischem Material, gefolgt von einem *Typen* („::“ = lexikalisch, „:“ = abgeleitet) und einer Liste syntaktischer Merkmale, wie sie im Abschnitt 2.1.1 diskutiert wurde. Ein *minimalistischer Ausdruck*, hinwiederum, ist eine Liste von Ketten. Die erste Kette in einem Ausdruck ist sein *Kopf*, der die strukturaufbauenden Funktionen „merge“ und „move“ steuert, die bei Ketten Fallunterscheidungen erforderlich machen. Diese lassen sich in Form von Schluss schemata angeben:

Seien s, t Sequenzen von Worten, f ein Merkmal, γ, δ Merkmalslisten, $\cdot \in \{::, :\}$ und α_i, β_j Ketten (i, j, k, l natürliche Zahlen), dann ist merge definiert durch die Schluss schemata:

$$\begin{array}{c}
\frac{s ::= f\gamma \quad t \cdot f, \alpha_1, \dots, \alpha_k}{st : \gamma, \alpha_1, \dots, \alpha_k} \text{ merge-1} \\
\frac{s := f\gamma, \alpha_1, \dots, \alpha_k \quad t \cdot f, \beta_1, \dots, \beta_l}{ts : \gamma, \alpha_1, \dots, \alpha_k, \beta_1, \dots, \beta_l} \text{ merge-2} \\
\frac{s \cdot = f\gamma, \alpha_1, \dots, \alpha_k \quad t \cdot f\delta, \beta_1, \dots, \beta_l}{s : \gamma, \alpha_1, \dots, \alpha_k, t : \delta, \beta_1, \dots, \beta_l} \text{ merge-3} .
\end{array}$$

Entsprechend ist move gegeben durch:

$$\begin{array}{c}
\frac{s : +f\gamma, \alpha_1, \dots, \alpha_{i-1}, t : -f, \alpha_{i+1}, \dots, \alpha_k}{ts : \gamma, \alpha_1, \dots, \alpha_{i-1}, \alpha_{i+1}, \dots, \alpha_k} \text{ move-1} \\
\frac{s : +f\gamma, \alpha_1, \dots, \alpha_{i-1}, t : -f\delta, \alpha_{i+1}, \dots, \alpha_k}{s : \gamma, \alpha_1, \dots, \alpha_{i-1}, t : \delta, \alpha_{i+1}, \dots, \alpha_k} \text{ move-2} .
\end{array}$$

Im Rahmen unserer MG-Modifikation definieren wir erstens die nichtkanonische Bewegung (s. Abb. 3):

$$\frac{s : +f\gamma, \alpha_1, \dots, \alpha_{i-1}, t : -f\delta, \alpha_{i+1}, \dots, \alpha_k}{ts : \delta\gamma, \alpha_1, \dots, \alpha_{i-1}, \alpha_{i+1}, \dots, \alpha_k} \text{ move-3} ,$$

wobei δ eine Folge von Lizensierern ist. Und zweitens die kovertierte Bewegung

$$\frac{s : +f\gamma, \alpha_1, \dots, \alpha_{i-1}, t : -f, \alpha_{i+1}, \dots, \alpha_k}{st : \gamma, \alpha_1, \dots, \alpha_{i-1}, \alpha_{i+1}, \dots, \alpha_k} \text{ move-4} ,$$

mit schwachen Merkmalen $+f, -f$.

Mit diesen Regeln lautet die Ableitung des Relativsatzes nach der von Hale (2003, 2006) ausgearbeiteten minimalistischen Promotionsanalyse:

$$\begin{array}{c}
\frac{\text{der} ::= \text{n} + \text{F d} - \text{nom} - \text{wh} \quad \text{Mann} ::= \text{n} - \text{f}}{\text{der} : + \text{F d} - \text{nom} - \text{wh}, \text{Mann} : - \text{f}} \text{ merge-3} \\
\frac{\text{der} : + \text{F d} - \text{nom} - \text{wh}, \text{Mann} : - \text{f}}{\text{Mann der} : \text{d} - \text{nom} - \text{wh}} \text{ move-1}
\end{array}$$

$$\begin{array}{c}
\frac{\text{ein} :: = n \ d \quad \text{Bart} :: n}{\text{ein Bart} : d} \text{ merge-1} \\
\frac{\varepsilon :: = d \ d \text{ --acc} \quad \text{ein Bart} : d}{\text{ein Bart} : d \text{ --acc}} \text{ merge-1} \\
\frac{\text{haben} :: = d \text{ +ACC } v \quad \text{ein Bart} : d \text{ --acc}}{\text{haben} : \text{ +ACC } v, \text{ ein Bart} : \text{ --acc}} \text{ merge-3} \\
\frac{\text{haben} : \text{ +ACC } v, \text{ ein Bart} : \text{ --acc}}{\text{einen Bart haben} : v} \text{ move-1} \\
\frac{(\text{pret}) :: V = = d \text{ +NOM } t \quad \text{einen Bart haben} : v}{\text{einen Bart hatte} : = d \text{ +NOM } t} \text{ merge-1'} \\
\frac{\text{einen Bart hatte} : = d \text{ +NOM } t \quad \text{Mann der} : d \text{ --nom --wh}}{\text{einen Bart hatte} : \text{ +NOM } t, \text{ Mann der} : \text{ --nom --wh}} \text{ merge-3} \\
\frac{\text{einen Bart hatte} : \text{ +NOM } t, \text{ Mann der} : \text{ --nom --wh}}{\text{einen Bart hatte} : t, \text{ Mann der} : \text{ --wh}} \text{ move-2} \\
\frac{\varepsilon :: = t \text{ +WH } c \quad \text{einen Bart hatte} : t, \text{ Mann der} : \text{ --wh}}{\text{einen Bart hatte} : \text{ +WH } c, \text{ Mann der} : \text{ --wh}} \text{ merge-1} \\
\frac{\text{einen Bart hatte} : \text{ +WH } c, \text{ Mann der} : \text{ --wh}}{\text{Mann der einen Bart hatte} : c} \text{ move-1} .
\end{array}$$

Für den korrekten Satz 1 lautet die Fortsetzung:

$$\begin{array}{c}
\frac{\text{ein} :: = c \ d \quad \text{Mann der einen Bart hatte} : c}{\text{ein Mann der einen Bart hatte} : d} \text{ merge-1} \\
\frac{k :: = d \ d \text{ --nom +npi} \quad \text{ein Mann der einen Bart hatte} : d}{\text{kein Mann der einen Bart hatte} : d \text{ --nom +npi}} \text{ merge-1} \\
\\
\frac{\text{sein} :: = a = \text{adv } v \quad \text{glücklich} :: a}{\text{sein glücklich} : = \text{adv } v} \text{ merge-1} \\
\frac{\text{sein glücklich} : = \text{adv } v \quad \text{jemals} :: \text{adv --npi}}{\text{sein glücklich} : v, \text{jemals} : \text{--npi}} \text{ merge-3} \\
\frac{(\text{pret}) :: V = = d \text{ +NOM } t \quad \text{sein glücklich} : v, \text{jemals} : \text{--npi}}{\text{war glücklich} : = d \text{ +NOM } t, \text{jemals} : \text{--npi}} \text{ merge-1} \\
\frac{\text{war glücklich} : = d \text{ +NOM } t, \text{jemals} : \text{--npi} \quad \text{kein Mann der einen Bart hatte} : d \text{ --nom +npi}}{\text{war glücklich} : \text{ +NOM } t, \text{jemals} : \text{--npi}, \text{kein Mann der einen Bart hatte} : \text{ --nom +npi}} \text{ merge-3} \\
\frac{\text{war glücklich} : \text{ +NOM } t, \text{jemals} : \text{--npi}, \text{kein Mann der einen Bart hatte} : \text{ --nom +npi}}{\text{kein Mann der einen Bart hatte war glücklich} : \text{ +npi } t, \text{jemals} : \text{--npi}} \text{ move-3} \\
\frac{\text{kein Mann der einen Bart hatte war glücklich} : \text{ +npi } t, \text{jemals} : \text{--npi}}{\text{kein Mann der einen Bart hatte war glücklich jemals} : t} \text{ move-4} \\
\frac{\varepsilon :: = t \ c \quad \text{kein Mann der einen Bart hatte war glücklich jemals} : t}{\text{kein Mann der einen Bart hatte war glücklich jemals} : c} \text{ merge-1} .
\end{array}$$

Dieser Satz hat Kategorie c und ist daher wohlgeformt.

Im Gegensatz dazu scheitert die Ableitung von Satz 3 daran, dass das NPI mit Merkmal $-npi$ nicht durch den korrespondierenden Lizenzierer $+npi$ lizenziert wird:

$$\begin{array}{c}
 \frac{\text{ein} :: c \ d \quad \text{Mann der einen Bart hatte} : c}{\text{ein Mann der einen Bart hatte} : d} \text{ merge-1} \\
 \frac{\varepsilon :: d \ d \ -nom \quad \text{ein Mann der einen Bart hatte} : d}{\text{ein Mann der einen Bart hatte} : d \ -nom} \text{ merge-1} \\
 \frac{\text{war glücklich} : d \ +NOM \ t, \text{jemals} : -npi \quad \text{ein Mann der einen Bart hatte} : d \ -nom}{\text{war glücklich} : +NOM \ t, \text{jemals} : -npi, \text{ein Mann der einen Bart hatte} : -nom} \text{ merge-3} \\
 \frac{\text{war glücklich} : +NOM \ t, \text{jemals} : -npi, \text{ein Mann der einen Bart hatte} : -nom}{\text{ein Mann der einen Bart hatte war glücklich} : t, \text{jemals} : -npi} \text{ move-1} .
 \end{array}$$

Interessanterweise scheitert die Ableitung von 2,

$$\begin{array}{c}
 \frac{\text{ein} :: n \ d \quad \text{Bart} :: n}{\text{ein Bart} : d} \text{ merge-1} \\
 \frac{k(neg) :: d \ d \ -acc \ +npi \quad \text{ein Bart} : d}{\text{kein Bart} : d \ -acc \ +npi} \text{ merge-1} \\
 \frac{\text{haben} :: d \ +ACC \ v \quad \text{kein Bart} : d \ -acc \ +npi}{\text{haben} :: +ACC \ v, \text{kein Bart} : -acc \ +npi} \text{ merge-3} \\
 \frac{\text{haben} :: +ACC \ v, \text{kein Bart} : -acc \ +npi}{\text{keinen Bart haben} : +npi \ v} \text{ move-3} ,
 \end{array}$$

daran, dass der Lizenzierer $+npi$ bereits den merger mit $(pret)::V = d \ +NOM \ t$ blockiert.

Literatur

- Anderson, J. A. & Rosenfeld, E., Hrsg. (1988). *Neurocomputing. Foundations of Research*, Bd. 1. MIT Press, Cambridge (MA).
- Anderson, J. R. (1996). *Kognitive Psychologie*. Spektrum, Heidelberg.
- Blutner, R. (2011). Taking a broader view: Abstraction and idealization. *Theoretical Linguistics*, 37(1-2):27–35.

- Braitenberg, V. & Pulvermüller, F. (1992). Entwurf einer neurologischen Theorie der Sprache. *Naturwissenschaften*, 79:103–117.
- Carstensen, K.-U., Ebert, C., Ebert, C., Jekat, S., Klabunde, R., & Langer, H., Hrsg. (2010). *Computerlinguistik und Sprachtechnologie*. Spektrum, Heidelberg.
- Christiansen, M. H. & Chater, N. (1999). Toward a connectionist model of recursion in human linguistic performance. *Cognitive Science*, 23(4):157–205.
- Christiansen, M. H., Chater, N., & Seidenberg, M. S., Hrsg. (1999). Sonderband *Connectionist Models of Human Language Processing*. *Cognitive Science*, 23(4).
- Churchland, P. S. & Sejnowski, T. J. (1997). *Grundlagen zur Neuroinformatik und Neurobiologie. The Computational Brain in deutscher Sprache*. Vieweg, Braunschweig.
- Crocker, M. W. (1996). *Computational Psycholinguistics*. Kluwer, Dordrecht.
- Delorme, A. & Makeig, S. (2004). EEGLAB: An open source toolbox for analysis of single-trial EEG dynamics including independent component analysis. *Journal of Neuroscience Methods*, 134:9–21.
- Dennett, D. C. (1978). *Intentional Systems*, chapter 1. MIT Press, Cambridge (MA). Nachdruck aus *Journal of Philosophy* 68(4), 87–106, 1971. Deutsch: „Intentionale Systeme“. In P. Bieri: *Analytische Philosophie des Geistes*, S. 162–183, 1997.
- Dowty, D. R., Karttunen, L., & Zwicky, A. M., Hrsg. (1985). *Natural Language Parsing. Psychological, Computational, and Theoretical Perspectives*. Cambridge University Press.
- Drenhaus, H. & beim Graben, P. (2012). Ereigniskorrelierte Potentiale (EKPs). *Zeitschrift für Germanistische Linguistik*, in diesem Band.
- Drenhaus, H., beim Graben, P., Saddy, D., & Frisch, S. (2006). Diagnosis and repair of negative polarity constructions in the light of symbolic resonance analysis. *Brain and Language*, 96 (3):255–268.
- Drenhaus, H., Frisch, S., & Saddy, D. (2005). Processing negative polarity items: When negation comes through the backdoor. In Kepser, S. & Reis, M., Hrsg., *Linguistic Evidence-Empirical, Theoretical, and Computational Perspectives*, S. 145–164, Berlin / New York. Mouton de Gruyter.
- Elman, J. L. (1990). Finding structure in time. *Cognitive Science*, 14:179–211.
- Elman, J. L. (1991). Distributed representations, simple recurrent networks, and grammatical structure. *Machine Learning*, 7:195–225.
- Elman, J. L. (1995). Language as a dynamical system. In Port, R. F. & van Gelder, T., Hrsg., *Mind as Motion: Explorations in the Dynamics of Cognition*, S. 195–223. MIT Press, Cambridge (MA).
- Elman, J. L., Bates, E. A., Johnson, M. H., Karmiloff-Smith, A., Parisi, D., & Plunkett, K. (1996). *Rethinking Innateness. A connectionist perspective on development*. Neural Network Modeling and Connectionism. MIT Press, Cambridge (MA).
- Farkas, I. & Crocker, M. W. (2008). Syntactic systematicity in sentence processing with a recurrent self-organizing network. *Neurocomputing*, 71:1172–1179.
- Friston, K. J. (1997). Transients, metastability, and neuronal dynamics. *NeuroImage*, 5(2):164–171.
- Frey, W. & Gärtner, H.-M. (2002). On the treatment of scrambling and adjunction in minimalist grammars. In Jäger, G., Monachesi, P., Penn, G., & Wintner, S., Hrsg., *Proceeding of Formal Grammar*, S. 41–52.
- Garagnani, M., Wennekers, T., & Pulvermüller, F. (2007). A neuronal model of the language cortex. *Neurocomputing*, 70:1914–1919.
- Gärtner, H.-M. & Jurish, B. (2011). Postmodern linguistics and the prospects of neural syntax: Some polemical remarks. *Theoretical Linguistics*, 37(1–2):37–44.

- Gärtner, H.-M. & Michaelis, J. (2005). A note on the complexity of constraint interaction: Locality conditions and minimalist grammars. In Blache, P., Stabler, E., Busquets, J., & Moot, R., Hrsg., *Logical Aspects of Computational Linguistics*, S. 3–10. Springer, Berlin.
- Gärtner, H.-M. & Michaelis, J. (2010). On the treatment of multiple-wh-interrogatives in minimalist grammars. In Hanneforth, T. and Fanselow, G., Hrsg., *Language and Logos*, S. 339–366. Akademie Verlag, Berlin.
- Gerth, S. & beim Graben, P. (2009). Unifying syntactic theory and sentence processing difficulty through a connectionist minimalist parser. *Cognitive Neurodynamics*, 3(4):297–316.
- beim Graben, P. (2004). Incompatible implementations of physical symbol systems. *Mind and Matter*, 2(2):29–51.
- beim Graben, P. (2008). Foundations of neurophysics. In beim Graben, P., Zhou, C., Thiel, M., & Kurths, J., Hrsg., *Lectures in Supercomputational Neuroscience: Dynamics in Complex Brain Networks*, Springer Complexity Series, Kapitel 1. Springer, Berlin.
- beim Graben, P., Gerth, S., & Vasisht, S. (2008a). Towards dynamical system models of language-related brain potentials. *Cognitive Neurodynamics*, 2(3):229–255.
- beim Graben, P. & Kurths, J. (2008). Simulating global properties of electroencephalograms with minimal random neural networks. *Neurocomputing*, 71(4):999–1007.
- beim Graben, P. & Potthast, R., Hrsg. (2009a). Sonderband *Language Dynamics*. *Cognitive Neurodynamics*, 3(4).
- beim Graben, P. & Potthast, R. (2009b). Inverse problems in dynamic cognitive modeling. *Chaos*, 19(1):015103.
- beim Graben, P., Zhou, C., Thiel, M., & Kurths, J., Hrsg. (2008b). *Lectures in Supercomputational Neuroscience: Dynamics in Complex Brain Networks*. Springer, Berlin.
- Gries, S. T. (2012). Statistische Modellierung. *Zeitschrift für Germanistische Linguistik*, in diesem Band.
- Hale, J. T. (2003). *Grammar, Uncertainty and Sentence Processing*. Dissertation, The Johns Hopkins University.
- Hale, J. T. (2006). Uncertainty about the rest of the sentence. *Cognitive Science*, 30(4):643–672.
- Harkema, H. (2001). *Parsing Minimalist Languages*. Dissertation, University of California, Los Angeles.
- Hebb, D. O. (1949). *The Organization of Behavior*. Wiley, New York (NY). Teilweise nachgedruckt in J. A. Anderson & E. Rosenfeld (1988), S. 45ff.
- Hinton, G. E. & Shallice, T. (1991). Lesioning an attractor network: Investigations of acquired dyslexia. *Psychological Review*, 98(1):74–95.
- Hofstadter, D. R. & Dennett, D. C., Hrsg. (1981). *Einsicht ins Ich: Fantasien und Reflexionen über Selbst und Seele*. Klett-Cotta, Stuttgart.
- Hopcroft, J. E. & Ullman, J. D. (1994). *Einführung in die Automatentheorie, formale Sprachen und Komplexitätstheorie*. Addison-Wesley, Bonn.
- Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences of the U.S.A.*, 79(8):2554–2558. Nachdruck in Anderson & Rosenfeld (1988), S. 460.
- Jäger, H. (1996). Dynamische Systeme in der Kognitionswissenschaft. *Kognitionswissenschaft*, 5:151–174.
- Joshi, A. K., Levy, L. S., & Takahashi, M. (1975). Tree adjunct grammars. *Journal of Computer and System Sciences*, 10(1):136–163.
- Joshi, A. K. & Schabes, Y. (1997). Treedjoining grammars. In Salomaa, A. & Rosenberg, G., Hrsg., *Handbook of Formal Languages and Automata*, S. 69–124. Springer, Berlin.

- Kandel, E. R., Schwartz, J. H., & Jessel, T. M., Hrsg. (1996). *Neurowissenschaften. Eine Einführung*. Spektrum der Wissenschaften, Akademischer Verlag, Heidelberg.
- Kaplan, D. & Glass, L. (1995). *Understanding Nonlinear Dynamics*. Springer, New York.
- Kawamoto, A. H. (1993). Nonlinear dynamics in the resolution of lexical ambiguity: A parallel distributed processing account. *Journal of Memory and Language*, 32:474–516.
- Kelso, J. A. S. (1995). *Dynamic Patterns. The Self-Organization of Brain and Behavior*. MIT Press, Cambridge (MA).
- Lawrence, S., Giles, C. L., & Fong, S. (2000). Natural language grammatical inference with recurrent neural networks. *IEEE Transactions on Knowledge and Data Engineering*, 12 (1):126–140.
- Lewis, R. L. (2003). Computational psycholinguistics. In *Encyclopedia of Cognitive Science*. Macmillan Reference Ltd., London.
- Liley, D. T. J., Cadusch, P. J., & Dafilis, M. P. (2002). A spatially continuous mean field theory of electrocortical activity. *Network: Computation in Neural Systems*, 13:67–113.
- Mainguy, T. (2010). A probabilistic top-down parser for minimalist grammars. arXiv:1010.1826 [cs.CL].
- McClelland, J. L. & Elman, J. L. (1986). The TRACE model of speech perception. *Cognitive Psychology*, 18(1):1–86.
- McCulloch, W. S. & Pitts, W. (1943). A logical calculus of ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics*, 5:115–133. Nachdruck in J. A. Anderson & E. Rosenfeld (1988), S. 18ff.
- Mizraji, E. (1989). Context-dependent associations in linear distributed memories. *Bulletin of Mathematical Biology*, 51(2):195–205.
- Näätänen, R., Paavilainen, P., Titinen, H., Jiang, D., & Alho, K. (1993). Attention and mismatch negativity. *Psychophysiology*, 30(5):436 – 450.
- Pereira, F. C. N. (1985). A new characterization of attachment preferences. In Dowty et al. (1985), 307–319.
- Plaut, D. C. (1996). Relearning after damage in connectionist networks: Toward a theory of rehabilitation. *Brain and Language*, 52(1):25–82.
- Plaut, D. C. & Shallice, T. (1993). Deep dyslexia: A case study of connectionist neuropsychology. *Cognitive Neuropsychology*, 10(5):377–500.
- Plunkett, K. & Elman, J. L. (1997). *Exercises in Rethinking Innateness. A Handbook for Connectionist Simulations*. Neural Network Modeling and Connectionism. MIT Press, Cambridge (MA).
- Potthast, R. & beim Graben, P. (2009). Inverse problems in neural field theory. *SIAM Journal on Applied Dynamical Systems*, 8(4):1405–1433.
- Rabinovich, M. I., Huerta, R., Varona, P., & Afraimovich, V. S. (2008). Transient cognitive dynamics, metastability, and decision making. *PLoS Computational Biology*, 4(5): e1000072.
- Rumelhart, D. E. & McClelland, J. L. (1986). On learning the past tenses of English verbs. In McClelland, J. L., Rumelhart, D. E., & the PDP research group, Hrsg., *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, Bd. II. MIT Press, Cambridge (MA).
- Schnelle, H. (1996). Approaches to computational brain theories of language. a review of recent proposals. *Theoretical Linguistics*, 22(1–2):49–104.
- Seidenberg, M. S. & McClelland, J. L. (1989). A distributed, developmental model of word recognition and naming. *Psychological Review*, 96:523–568.

- Sejnowski, T. J. & Rosenberg, C. R. (1987). Parallel networks that learn to pronounce English text. *Complex Systems*, 1:145 – 168.
- Shieber, S. M. (1985). Evidence against the context-freeness of natural language. *Linguistics and Philosophy*, 8:333–343.
- Siegelmann, H. T. & Sontag, E. D. (1995). On the computational power of neural nets. *Journal of Computer and System Sciences*, 50(1):132–150.
- Smolensky, P. (1988). On the proper treatment of connectionism. *Behavioral and Brain Sciences*, 11(1):1–74.
- Smolensky, P. (1990). Tensor product variable binding and the representation of symbolic structures in connectionist systems. *Artificial Intelligence*, 46(1–2):159–216.
- Smolensky, P. (2006). Harmony in linguistic cognition. *Cognitive Science*, 30:779–801.
- Smolensky, P. & Legendre, G. (2006a). *The Harmonic Mind. From Neural Computation to Optimality-Theoretic Grammar*, Bd. 1: Cognitive Architecture. MIT Press, Cambridge (MA).
- Smolensky, P. & Legendre, G. (2006b). *The Harmonic Mind. From Neural Computation to Optimality-Theoretic Grammar*, Bd. 2: Linguistic and Philosophic Implications. MIT Press, Cambridge (MA).
- Stabler, E. P. (1997). Derivational minimalism. In Retoré, C., Hrsg., *Logical Aspects of Computational Linguistics*, S. 68–95. Springer, New York.
- Stabler, E. P. (2004). Varieties of crossing dependencies: structure dependence and mild context sensitivity. *Cognitive Science*, 28:699–720.
- Stabler, E. P. (2011a). Computational perspectives on minimalism. In Boeckx, C., Hrsg., *Oxford Handbook of Linguistic Minimalism*, S. 617–641. Oxford University Press.
- Stabler, E. P. (2011b). Top-down recognizers for MCFGs and MGs. In *Proceedings of the 2nd Workshop on Cognitive Modeling and Computational Linguistics*, S. 39–48, Portland, Oregon, USA. Association for Computational Linguistics.
- Stabler, E. P. & Keenan, E. L. (2003). Structural similarity within and among languages. *Theoretical Computer Science*, 293:345–363.
- Tabor, W. (2000). Fractal encoding of context-free grammars in connectionist networks. *Expert Systems: The International Journal of Knowledge Engineering and Neural Networks*, 17(1):41–56.
- Tabor, W., Juliano, C., & Tanenhaus, M. K. (1997). Parsing in a dynamical system: An attractor-based account of the interaction of lexical and structural constraints in sentence processing. *Language and Cognitive Processes*, 12(2/3):211–271.
- Tabor, W. & Tanenhaus, M. K. (1999). Dynamical models of sentence processing. *Cognitive Science*, 23(4):491–515.
- Turing, A. M. (1950). Computing machinery and intelligence. *Mind*, 59(236). Teilweise nachgedruckt in Hofstadter, D. R. & Dennett, D. C. (1981).
- Vasisht, S., Brüßow, S., Lewis, R. L., & Drenhaus, H. (2008). Processing polarity: How the ungrammatical intrudes on the grammatical. *Cognitive Science*, 32(4):685–712.
- Vosse, T. & Kempen, G. (2000). Syntactic structure assembly in human parsing: a computational model based on competitive inhibition and a lexicalist grammar. *Cognition*, 75:105–143.
- Vosse, T. & Kempen, G. (2009). The Unification Space implemented as a localist neural net: Predictions and error-tolerance in a constraint-based parser. *Cognitive Neurodynamics*, 3(4):331–346.
- Wennekers, T., Garagnani, M., & Pulvermüller, F. (2006). Language models based on Hebbian cell assemblies. *Journal of Physiology*, 100:16–30.